PLFS Update

LANL: John Bent, HB Chen, David Gunter, Gary Grider, Sam Gutierrez, Adam Manzanares, Ben McClelland, Dave Montoya, James Nunez, Alfred Torrez, Meghan Wingate

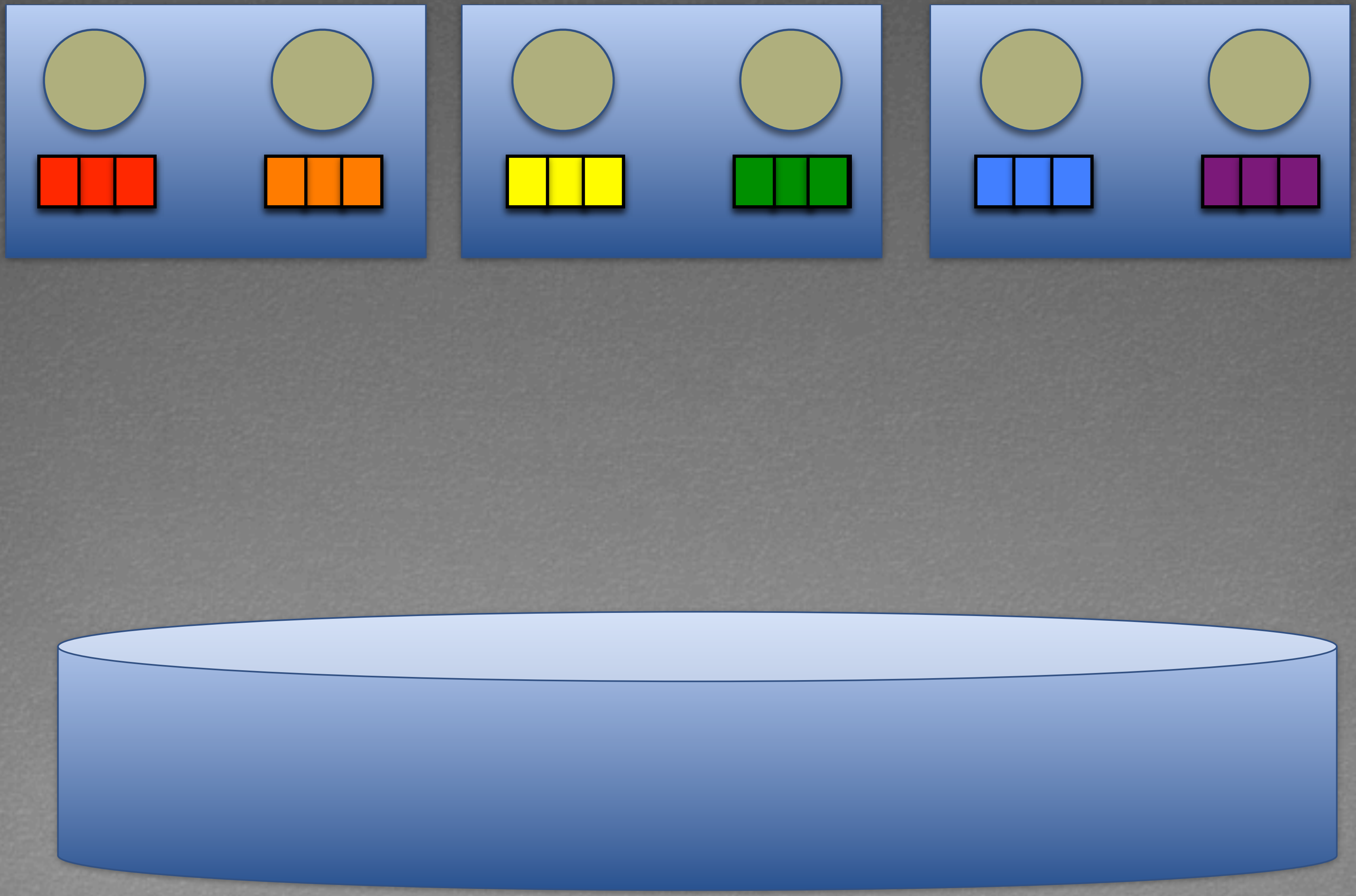
CMU: Garth Gibson, Milo Polte

PSC: Paul Nowoczinski

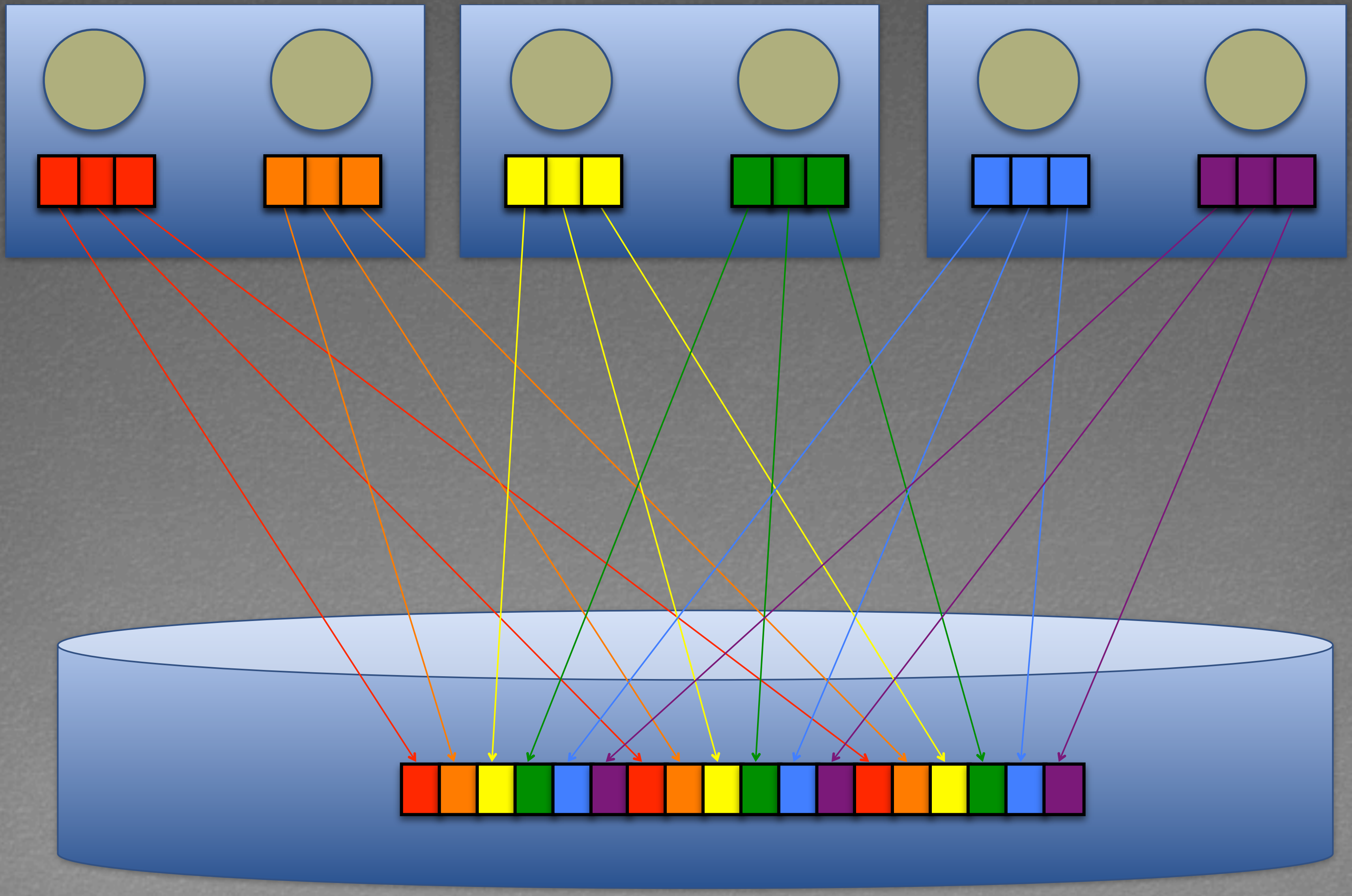
PLFS Overview

- Virtual parallel file system
- Interposes between application and another physical parallel file system
- Reorganizes application IO
  - From N-1 into N-N

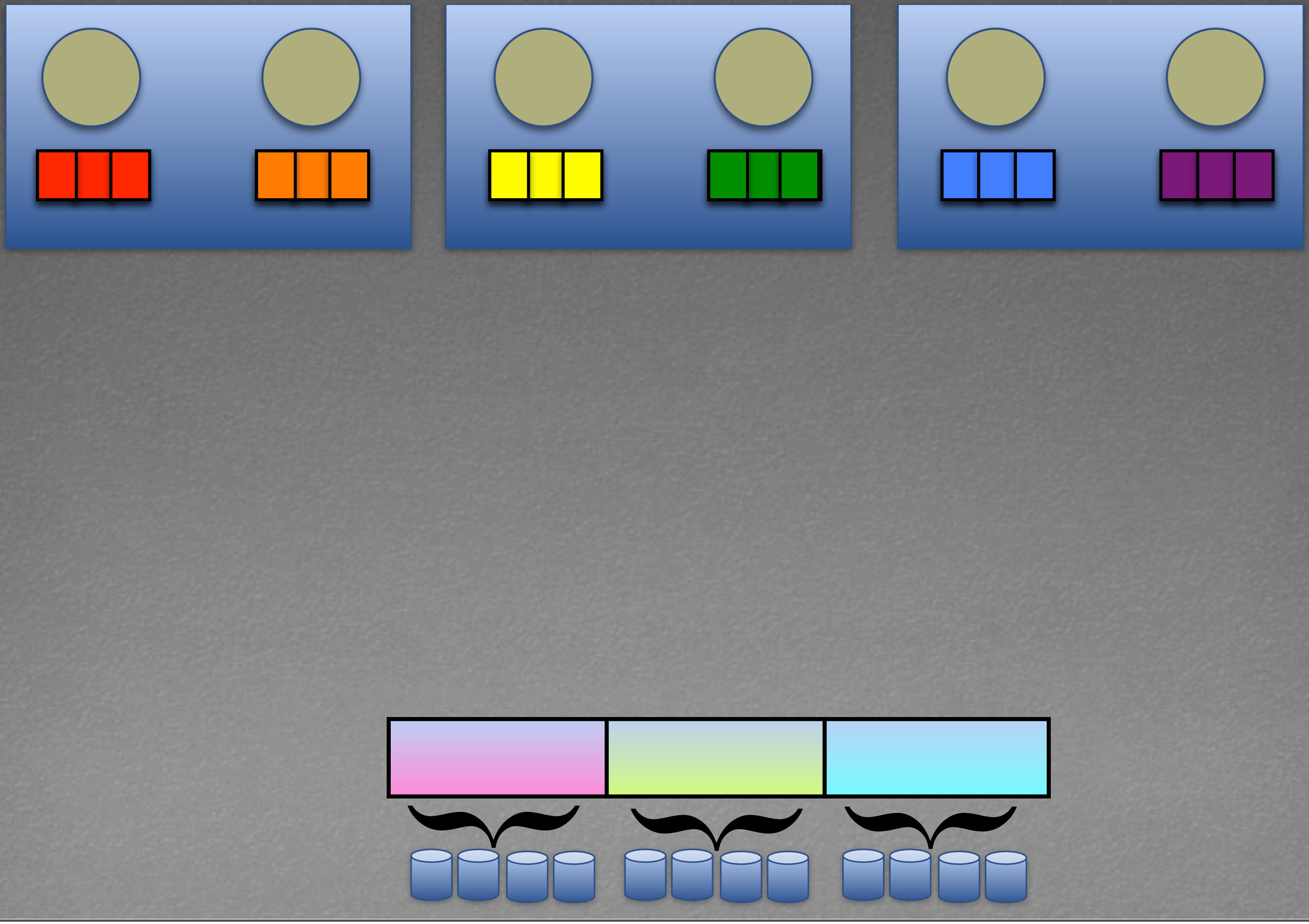
Challenge of N-1 Strided



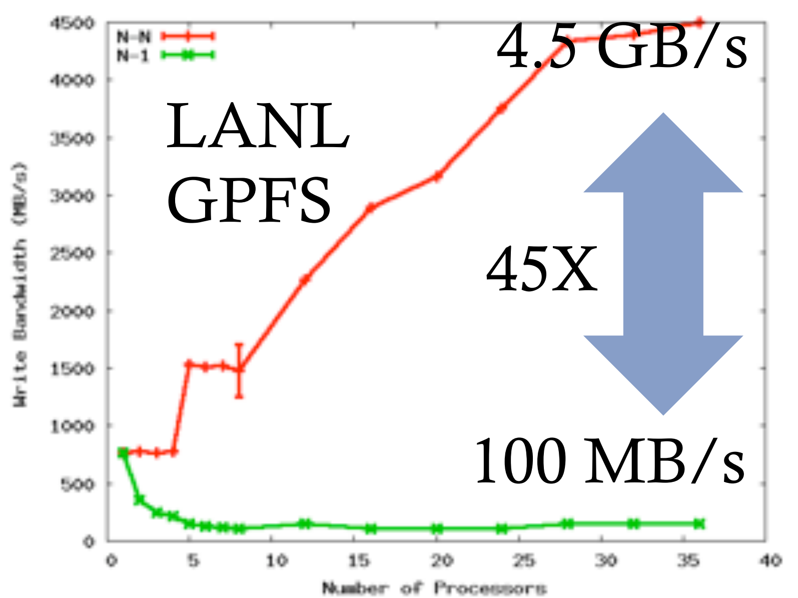
Challenge of N-1 Strided



Challenge of N-1 Strided



N-1 Performance Penalty

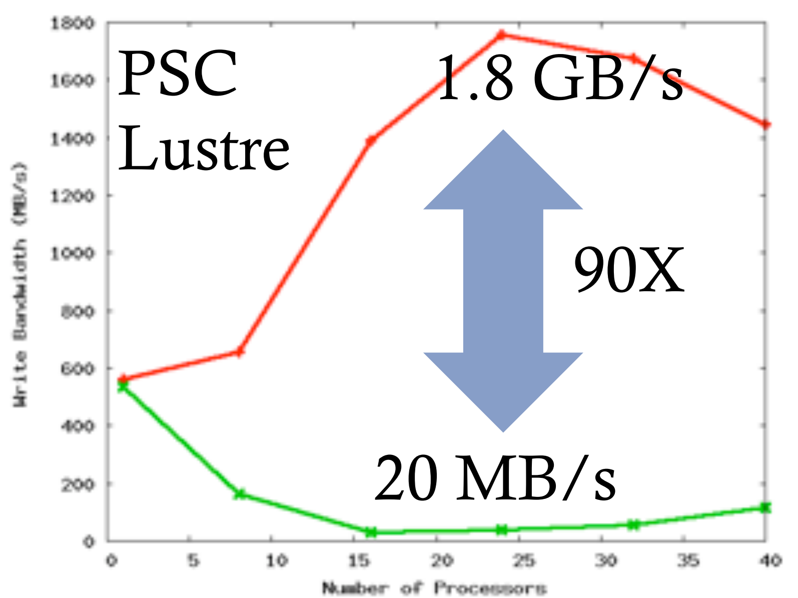


LANL GPFS

4.5 GB/s

45X

100 MB/s

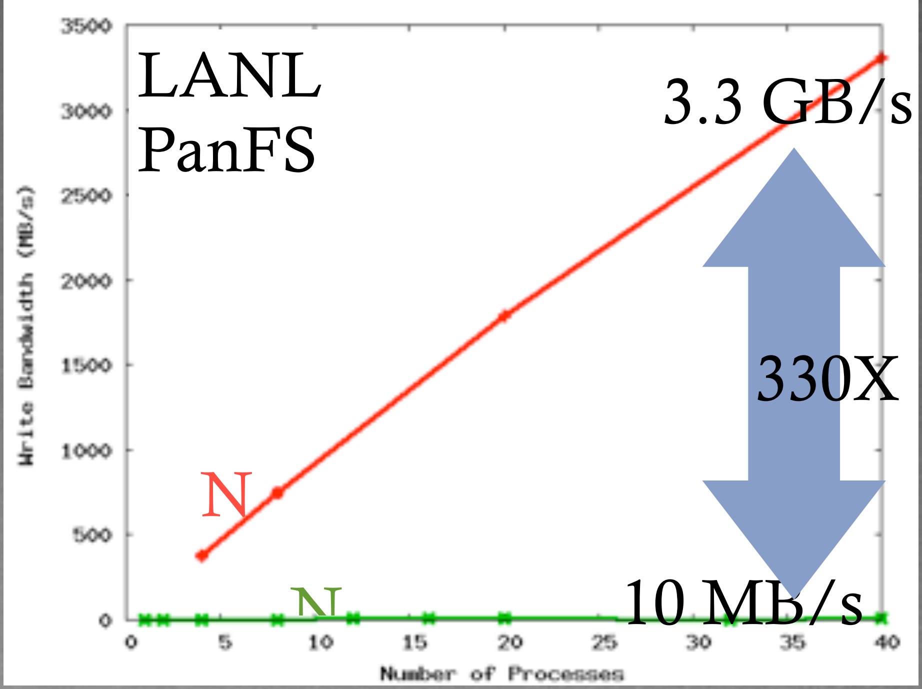


PSC Lustre

1.8 GB/s

90X

20 MB/s



LANL PanFS

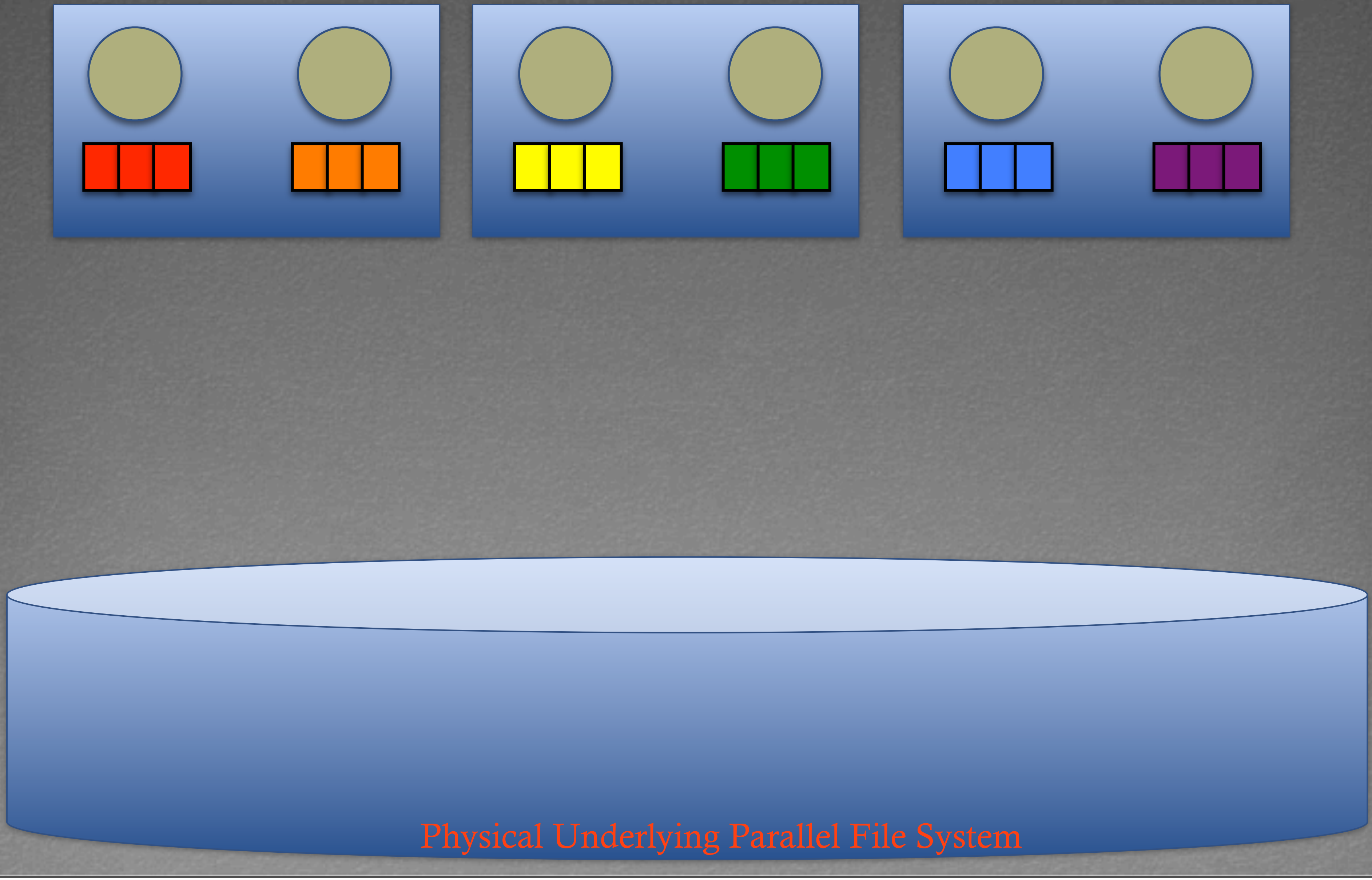
3.3 GB/s

330X

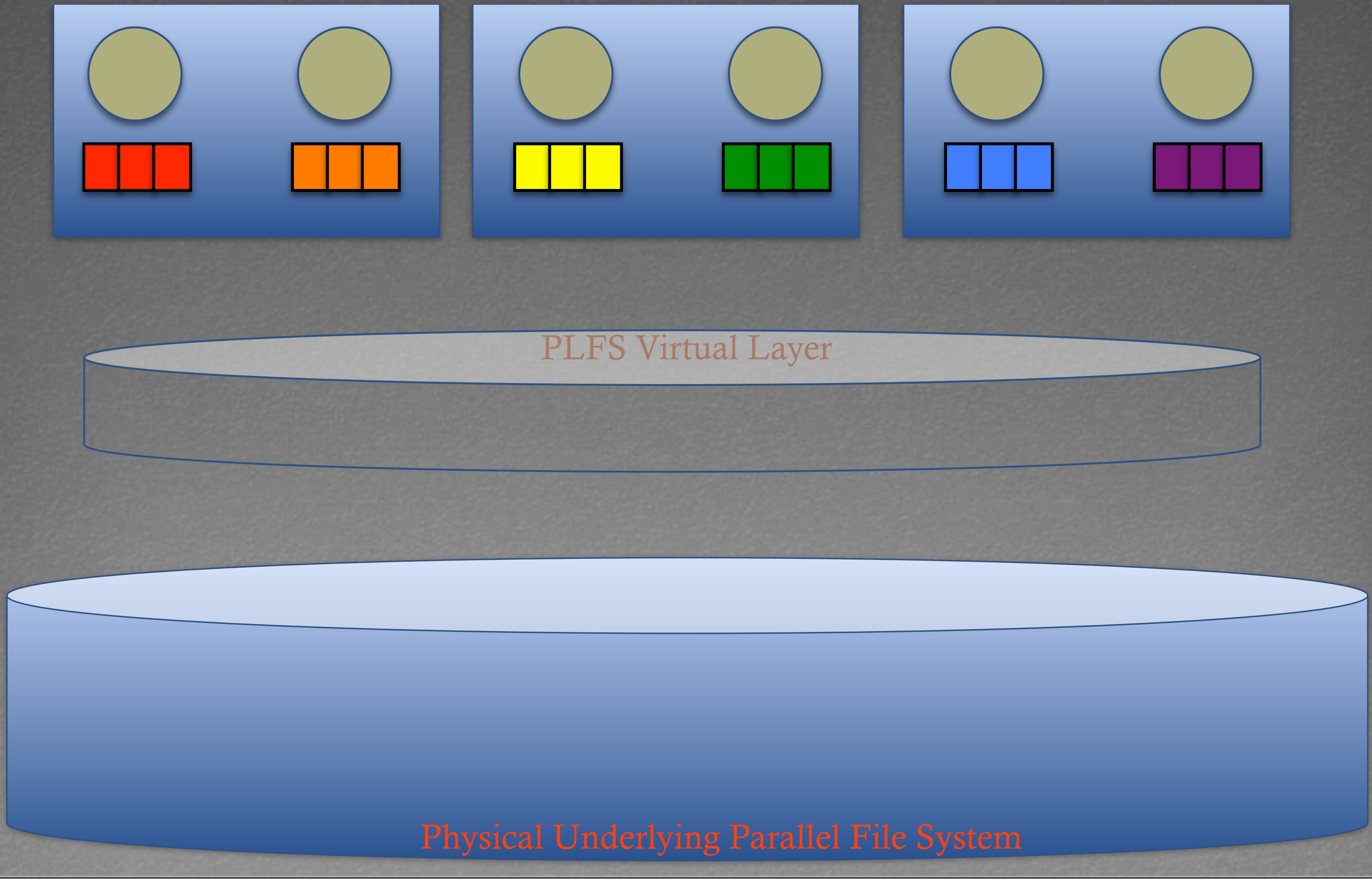
10 MB/s

N-N
N-1

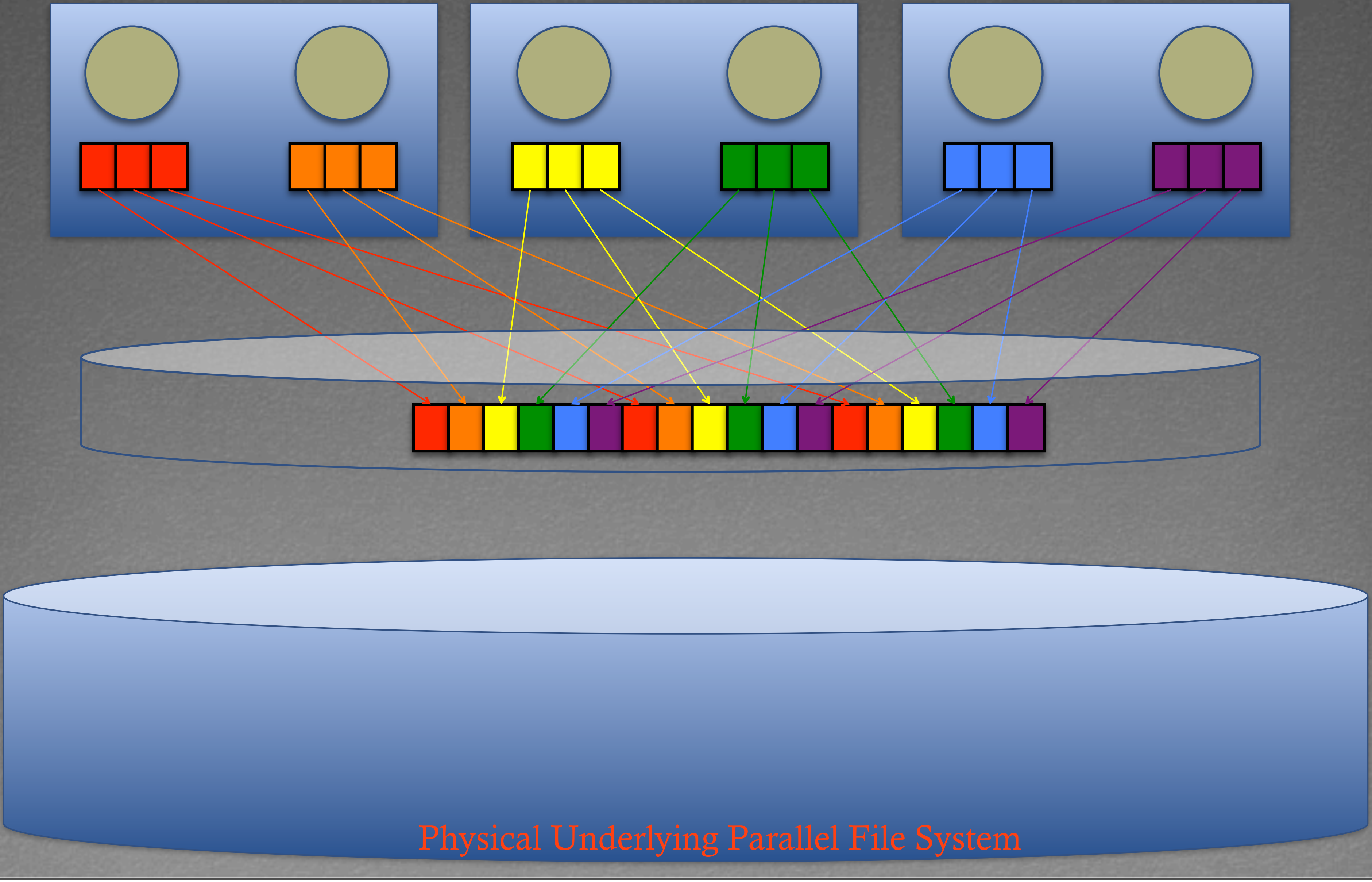
PLFS High Level



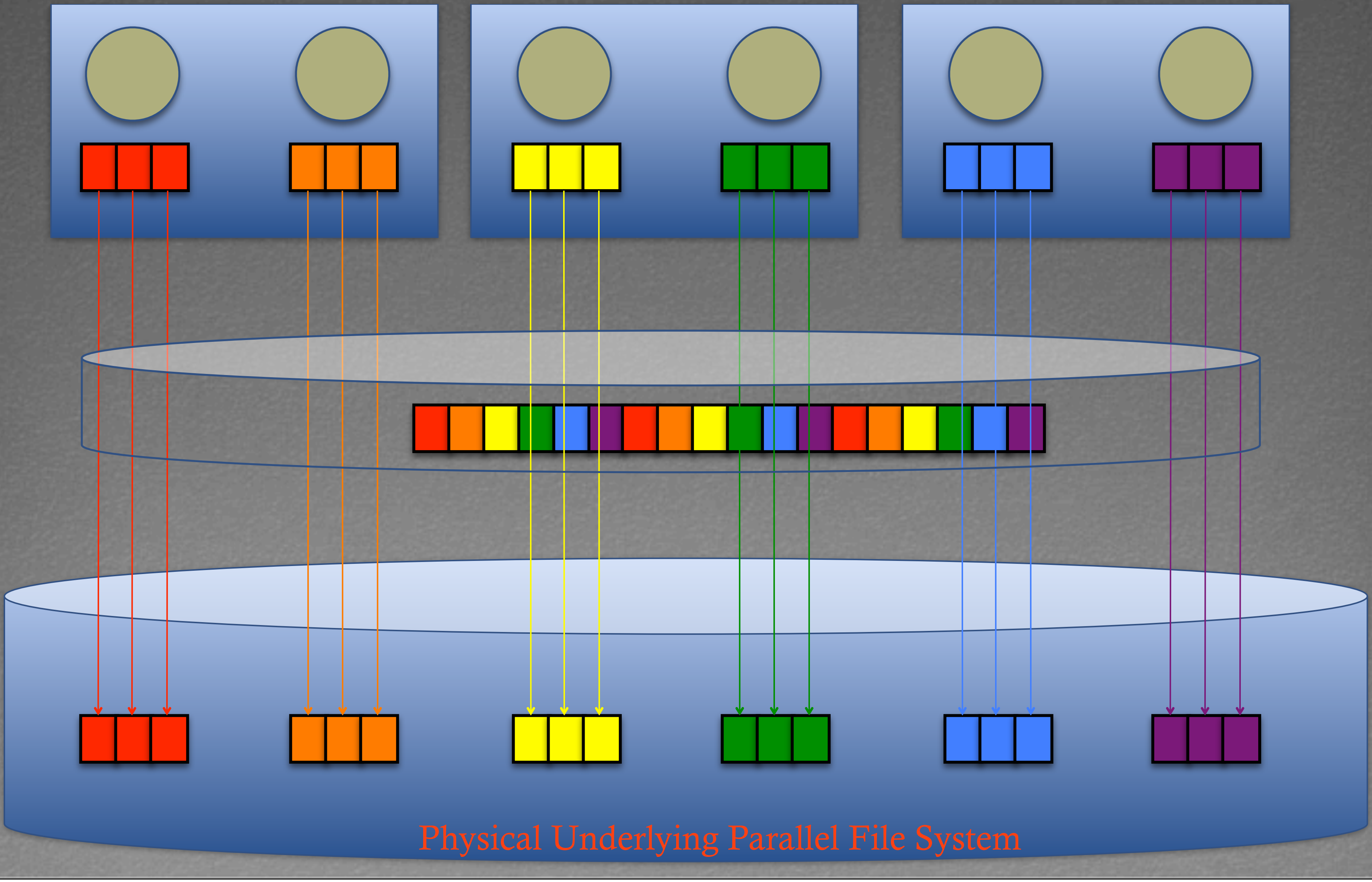
PLFS High Level



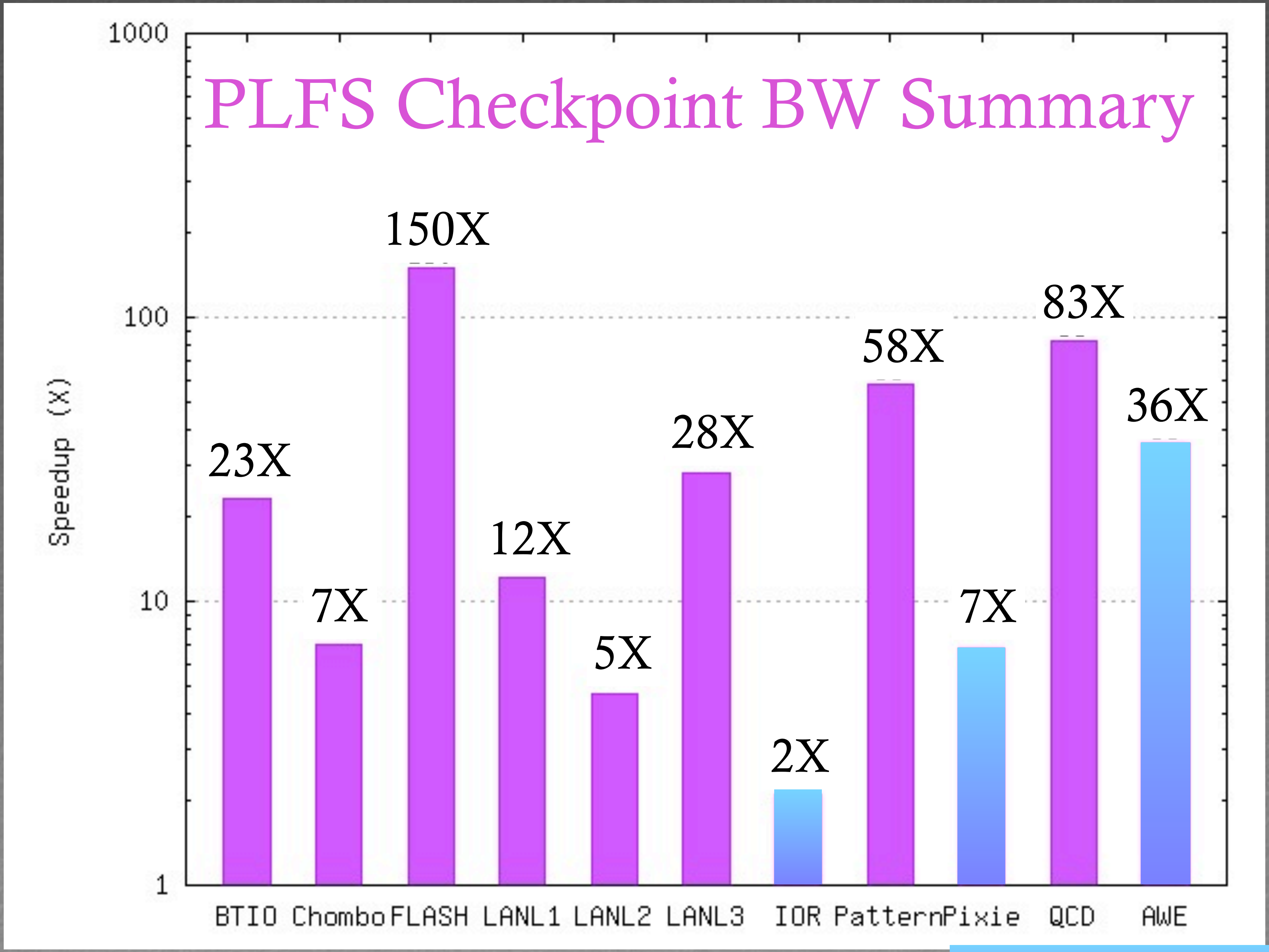
PLFS High Level



PLFS High Level



PLFS Checkpoint BW Summary

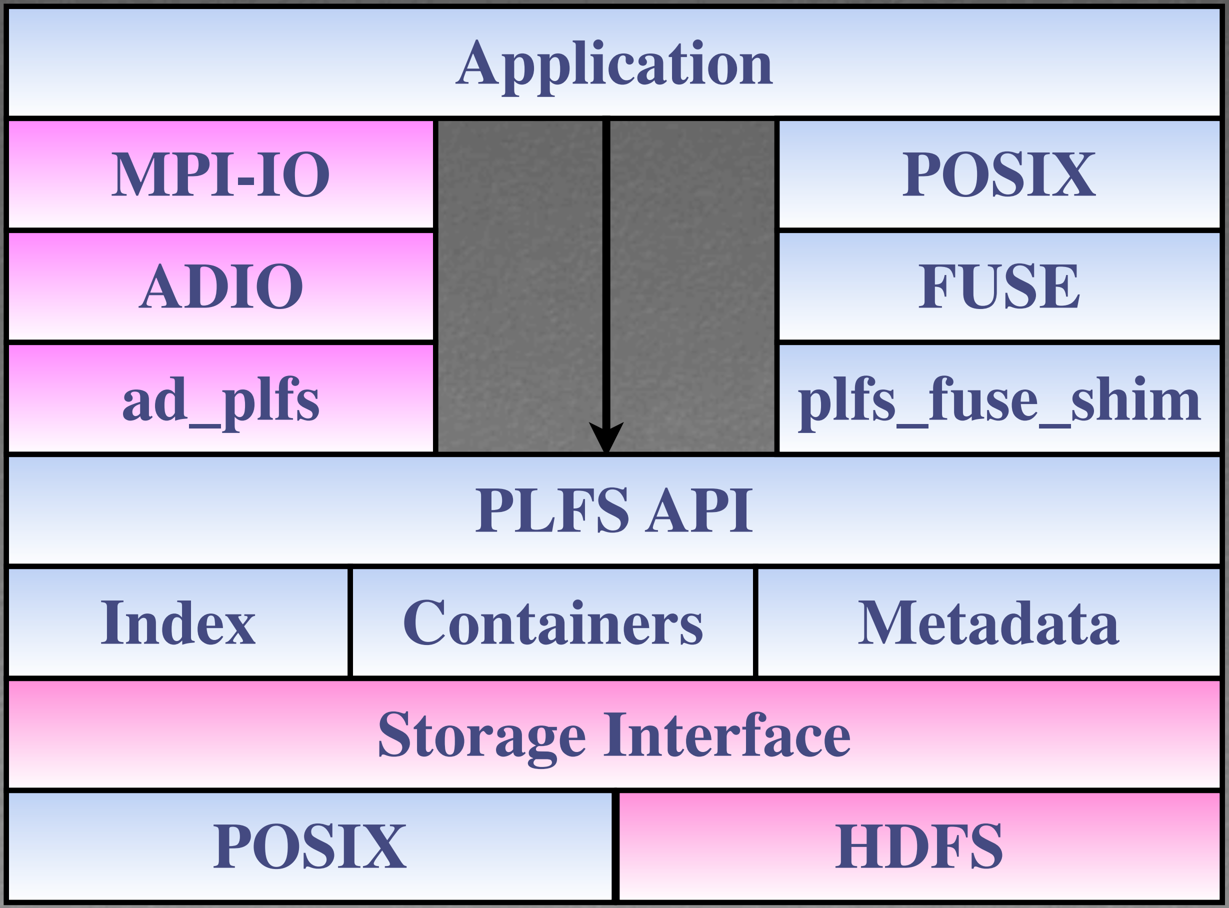


New Results since SC09

PLFS Status:
Production Version 1.0 imminent

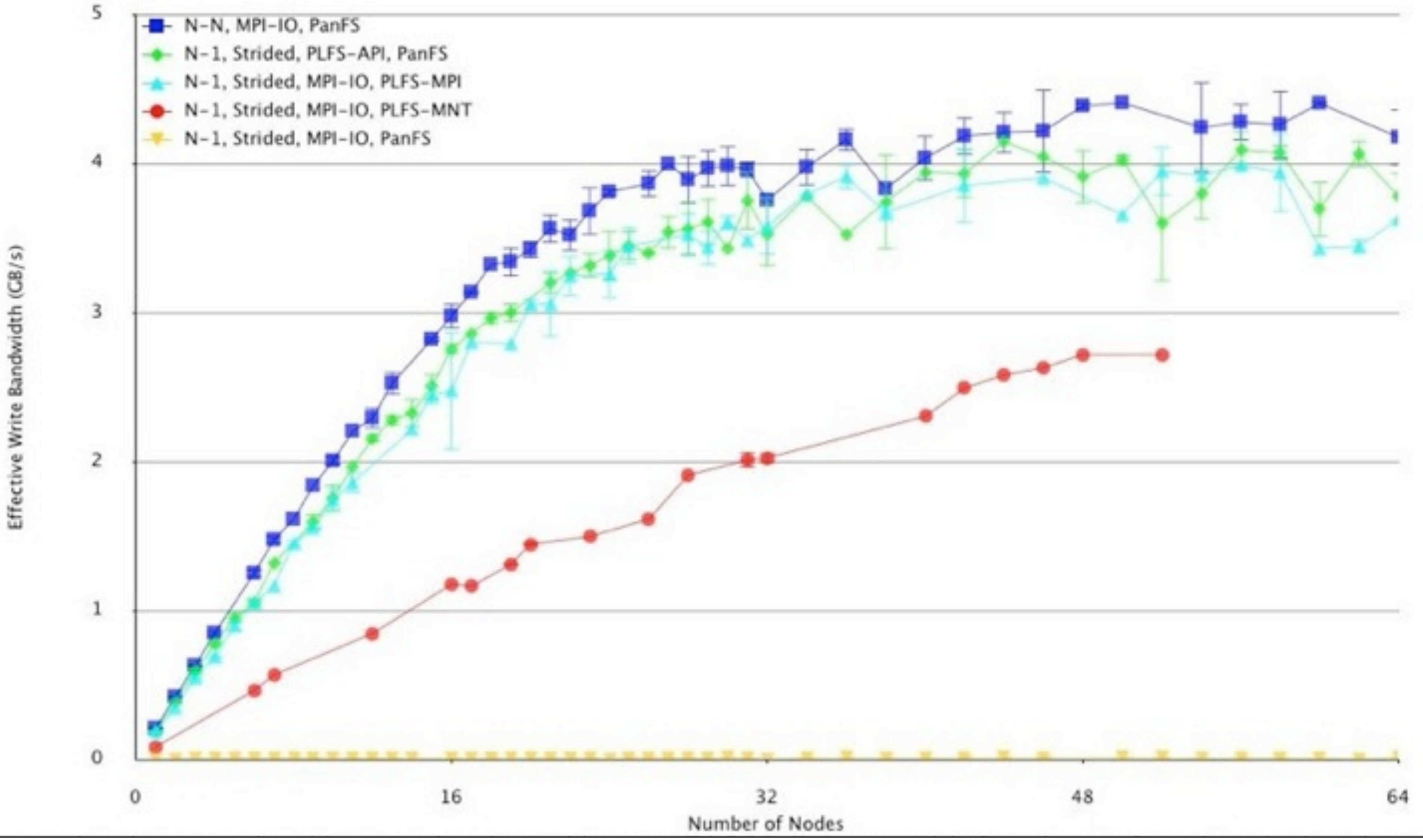
- Current large push at LANL into production
  - Running on Roadrunner for friendly users
  - Big debug push to get onto Roadrunner for all users
  - And other supercomputers as well: current and future
- New features needed for Version 1.0 production release
  - MPI-IO interface
  - Thread-pools for archiving workloads

PLFS Stack



New Bits

Multiple PLFS interfaces (MPI-IO, POSIX, API)

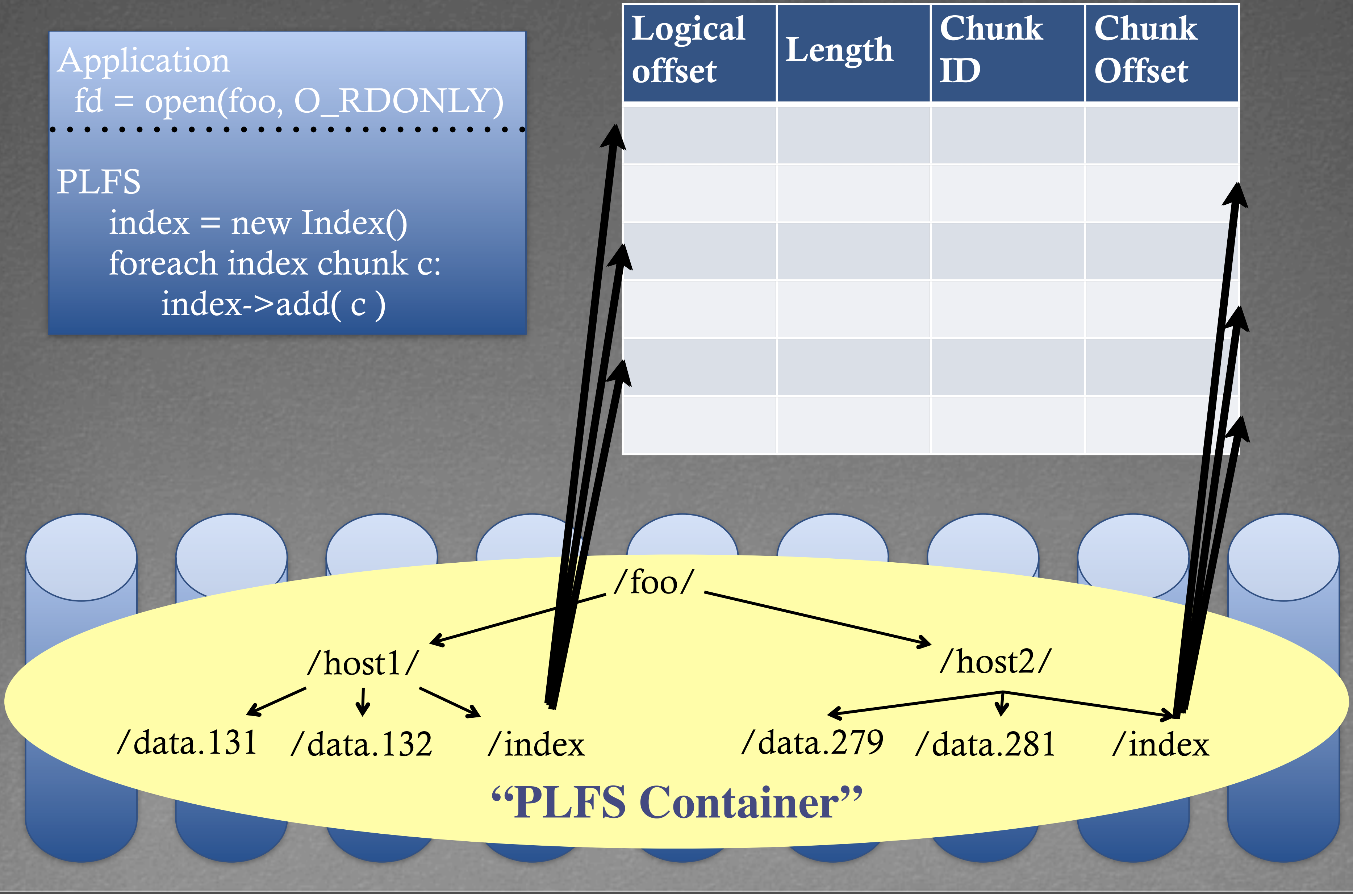


Write-optimized, now discovering read issues

- Read open time for files created by lots of writers
- Large sequential read bandwidth for files created with lots of small writes

open() for read

```
Application
  fd = open(foo, O_RDONLY)
. . . . . . . . . . . . . . . . . . . . . . . . . . .
PLFS
    index = new Index()
    foreach index chunk c:
        index->add( c )
```



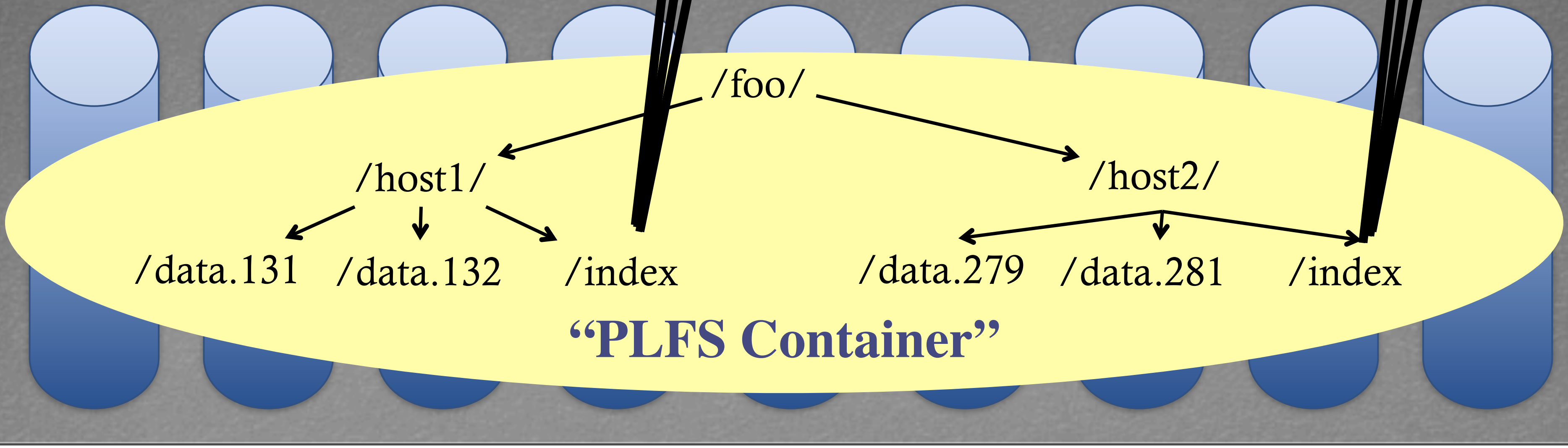
Logical offset	Length	Chunk ID	Chunk Offset

open() for read

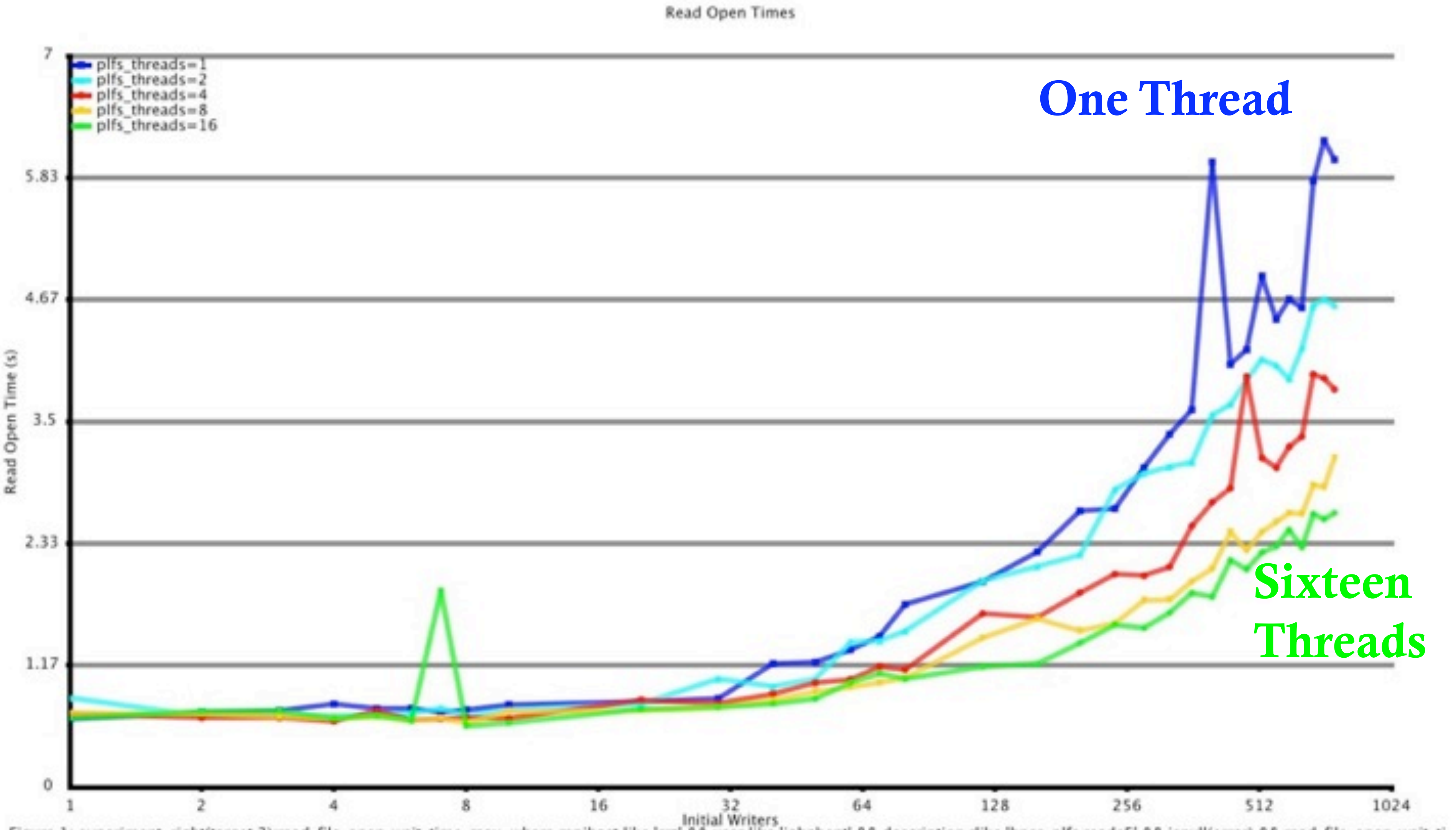
```
Application
  fd = open(foo, O_RDONLY)
PLFS
    index = new Index()
    foreach index chunk c:
        index->add( c )
```

Replace foreach with threads

Logical offset	Length	Chunk ID	Chunk Offset



Improving Read Open Times



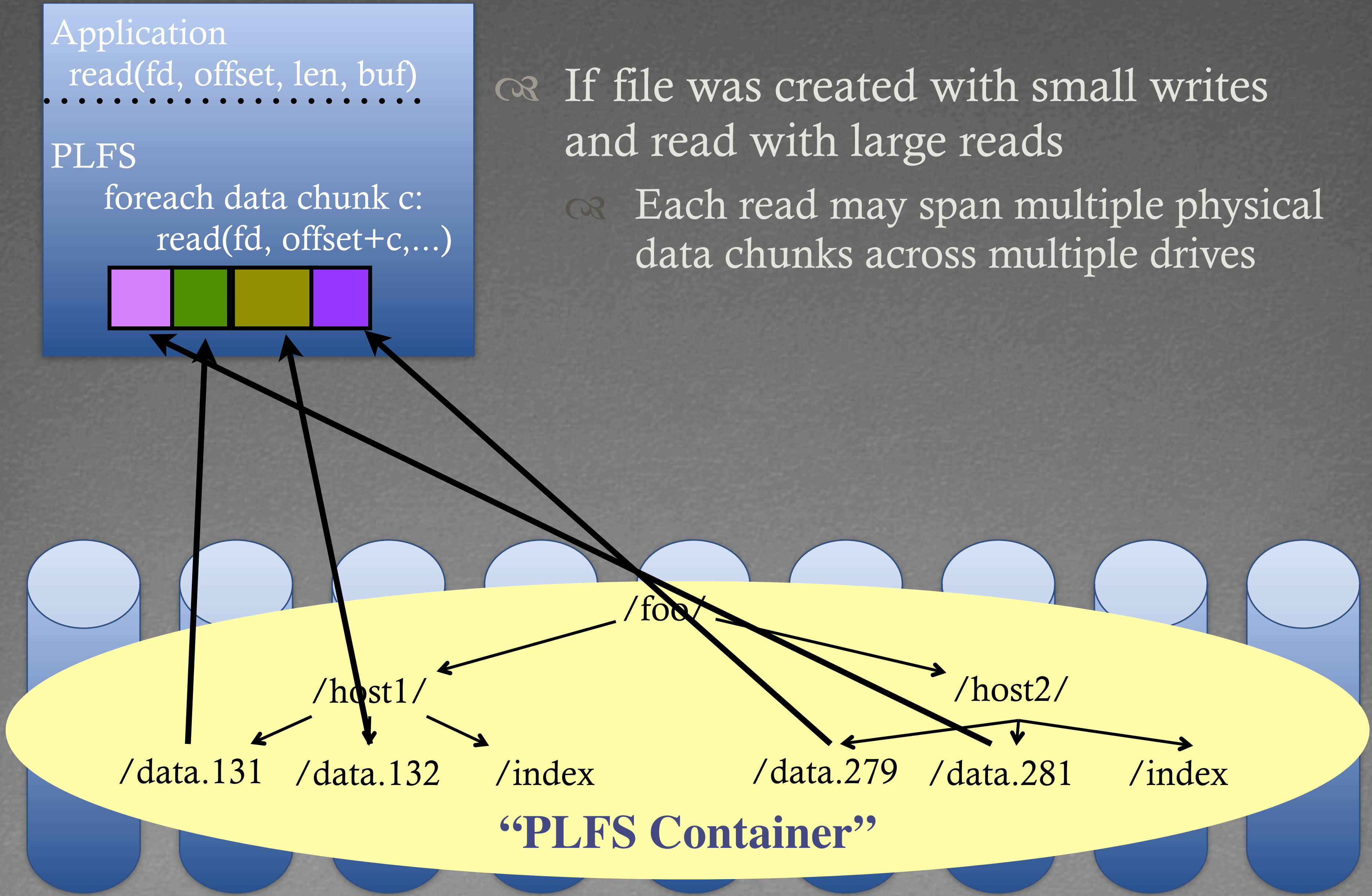
One Thread

Sixteen Threads

Figure 1: experiment, right(target,3):read_file_open_wait_time_max, where mpihost like 'rrz' && user like 'johnbent' && description rlike 'hpss_plfs.reads5' && isnull(error) && read_file_open_wait_time_max < 20 && plfs_threads<=16 && total_size_mb > 20*1024 (1405 rows, 1280857844)

Improving Read Bandwidths

- If file was created with small writes and read with large reads
  - Each read may span multiple physical data chunks across multiple drives



Application

```
read(fd, offset, len, buf)
```

PLFS

```
foreach data chunk c:
    read(fd, offset+c,...)
```

/foo/

/host1/

/data.131 /data.132 /index

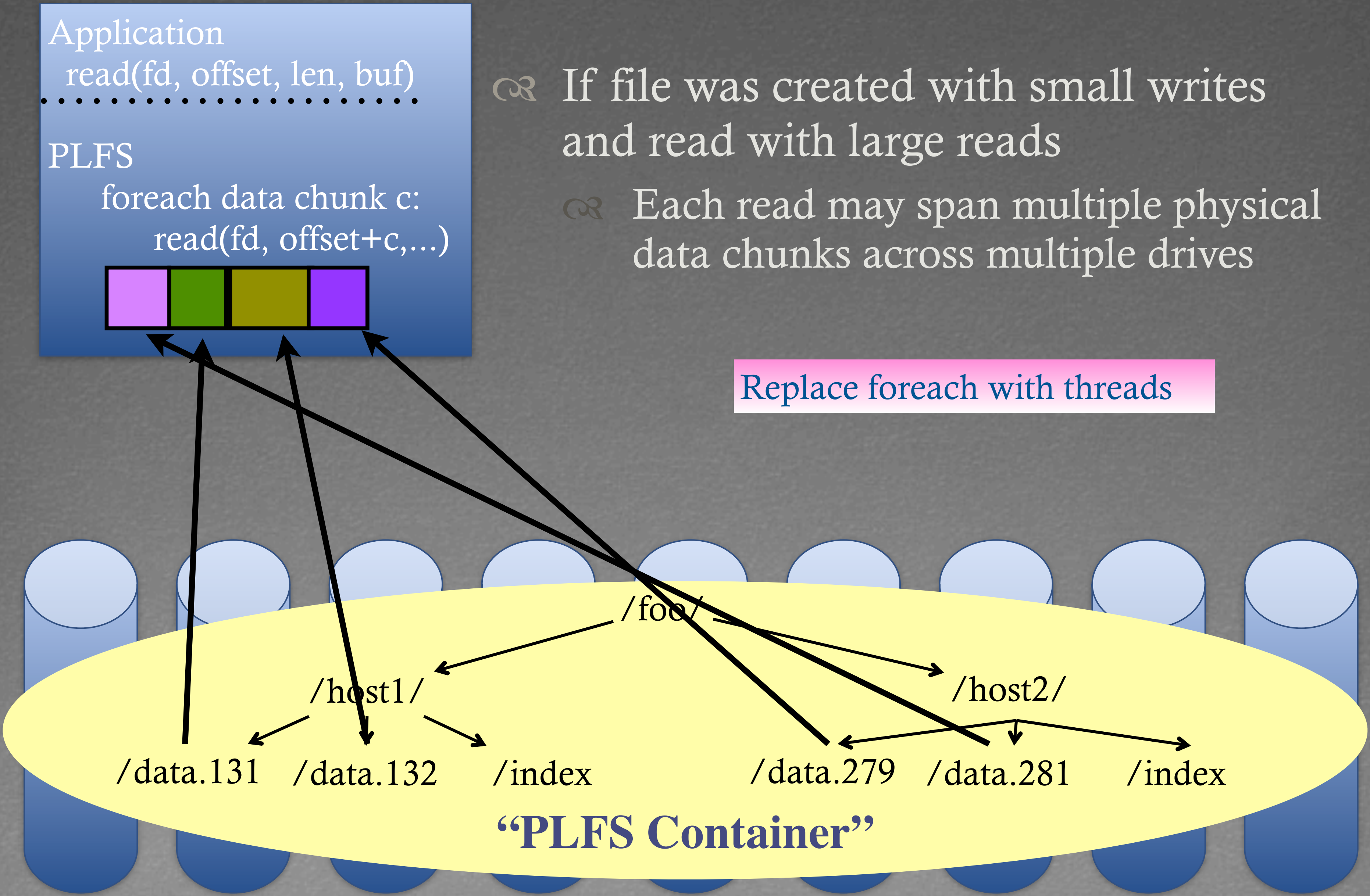
/host2/

/data.279 /data.281 /index

“PLFS Container”

Improving Read Bandwidths

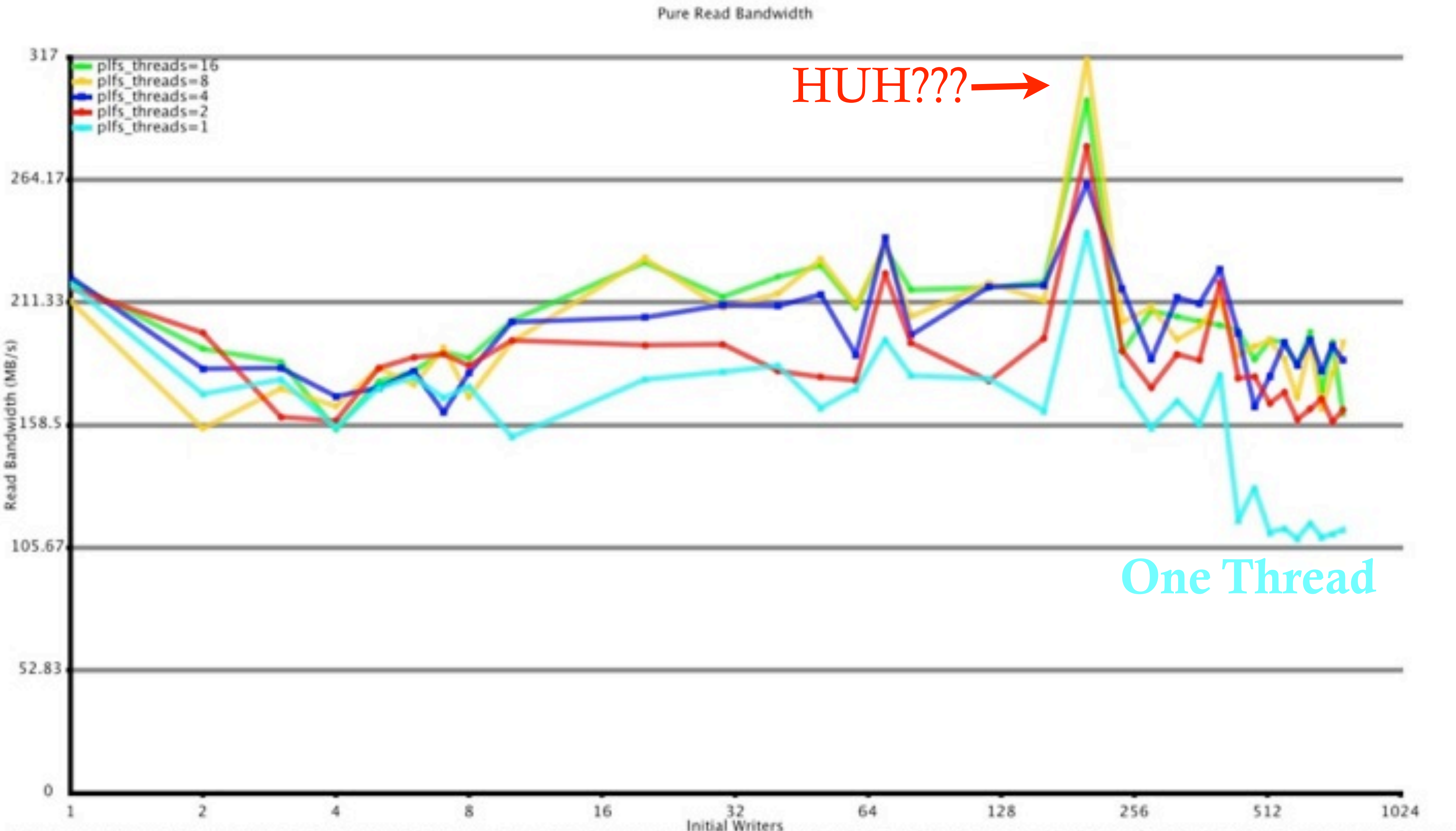
- If file was created with small writes and read with large reads
  - Each read may span multiple physical data chunks across multiple drives



```
Application
  read(fd, offset, len, buf)
PLFS
    foreach data chunk c:
        read(fd, offset+c,...)
```

Replace foreach with threads

Multi-Threaded Reads



HUH???→

One Thread

Figure 2: experiment, right(target,3):total_size_mb/read_total_op_time_max, where mpihost like 'rrz' && user like 'johnbent' && description rlike 'hpss_plfs.reads5' && isnull(error) && read_file_open_wait_time_max < 20 && plfs_threads<=16 && total_size_mb > 20*1024 (1405 rows, 1280857848)

Conclusion

- Version 1.0 imminent
- Open source
  - http://sourceforge.net/projects/plfs/
  - > svn co https://plfs.svn.sourceforge.net/svnroot/plfs plfs
- Collaborations, contributions, bug reports very welcome
- Version 2.0 not imminent
  - Further virtualization and re-organization of user data
  - Distribute metadata workload transparently
  - Use communicators in MPI to optimize PLFS ADIO layer